Data Mining on NASA's Information Power Grid

By Thomas H. Hinke* and Jason Novotny+
*Ames Research Center (on sabbatical leave from the University of Alabama in Huntsville)
+National Center for Supercomputing Applications

Abstract

This paper describes the development of a data mining system that is to operate on NASA's Information Power Grid (IPG). Mining agents will be staged to one or more processors on the IPG. There they will grow using just-in-time acquision of new operations. They will mine data delivered using just-in-time delivery. Some initial experimental results are presented.

1. Introduction

This paper describes the development of a data mining system that is to operate on NASA's Information Power Grid (IPG), which is built using the Globus toolkit.[1] The data mining system targets the mining of remotely sensed satellite data, which is characterized by its potentially large volume. A definition of data mining from a recent NASA data mining workshop states that "Data mining is the process by which information and knowledge are extracted from a potentially large volume of data using techniques that go beyond a simple search through the data." [2] This paper represents a snap-shot into a project that is ongoing, presenting a scenario of grid-based mining, an architecture for a grid-based miner, and some initial experimental results.

2. Scenario of IPG Mining

The user must specify *what* is to be mined, *how* it is to be mined and *where* it is to be mined. Under our approach, the user specifies *what* is to be mined by storing the names and grid locations of a set of data in a database associated with the IPG miner. These are communicated to a miner agent over the grid. The user specifies *how* the data is to be mined by specifying a mining plan that lists the sequence of mining operations that are to be applied to the data and any parameters required. The user specifies *where* the mining is to take place by specifying the IPG processors on which the mining agent is to be staged. With these requirements specified, the user will then invoke the miner, which will send mining agents to the designated IPG processors. On these processors, each agent will acquire the data to be mined, mine it and send the results back to the user.

3. IPG Data Mining Architecture

The starting point for this development was the ADaM data mining system that was developed at the University of Alabama in Huntsville under a NASA grant.[3,4,5] The ADaM data mining system was developed to be extensible, using an object-oriented design that was coded in C++.

To begin the mining operation, initially a "thin" mining agent and its associated mining plan will be staged to an IPG processor. These thin agents will grow through the acquisition of the necessary mining operations required to execute the plan. It is envisioned that these thin agent may acquire mining operations from multiple sites on the IPG. Some will be acquired from public repository sites that contain a standard set of mining operations. It is hoped that once the mining system is fully operational, mining users will contribute new operations to this mining repository. Proprietary mining operations could be acquired from the user's private set of mining operations or from companies that might, in the future, sell mining operations. Because of the multitude of sources for the mining operations, it was felt that this "just in time" mining operation acquisition approach would represent a reasonable initial design strategy for the IPG miner.

The agent also performs "just-in- time" acquisition of data. Such data can be acquired from IPG-based repositories as well as various NASA data repositories that provide FTP access to their data holdings. By using just-in-time data delivery, the storage requirments for the target mining site are minimized.

The IPG mining architecture supports coarse-grained parallelism. An alternative approach would support fine-grained parallelism, in which a portion of a mining plan would be performed on one processor, and then partial results would be sent on to another processor for additional work, in a pipe-lined approach. This approach is not currently supported, because of the high data volumes involved and the overhead of transmitting data across the grid. Some type of partial result sharing could, however, be involved in future work in which data mining systems working on different data sets may want to collaborate by sharing partial results.

5. Initial Experiments

We performed a very simple experiment in which one day of SSM/I data was mined. [6] For this experiment, the data mining system used just-in-time delivery to acquired one day (14 orbits) of SSM/I data. This data (consisting of 75 megabytes of data organized into 84 files) was acquired from a remote host using both FTP and Globus transport mechanisms. As a reference point, the IPG miner was also used to mine the same data that was stored locally. The Unix csh time command was used to time the commands, with its wall-clock time reported in the following table. These are preliminary results, since they represent only one run for each experiment, with no optimization in the just-in-time delivery. There is no overlap between mining and the receipt of the next file. However, these results provide some indication of the performance of the IPG data mining system.

Source of Data	**Time to Acquire and Mine Data**
Using Globus from Remote Host	**8 minutes 59 seconds**
Using FTP from Remote Host	**8 minutes 32 seconds**
Local Host	**5 minutes 22 seconds**

6. Future Work

While the system has been used to spawn a swarm of agents running on the NASA Ames 512 processor SGI, some additional work is required to permit the agents to operate autonomously within the wider IPG environment as described in the architecture section. Work also needs to be completed to permit the user to monitor the progress of swarms of agents operating on a set of processors within the IPG. Finally, additional work needs to be done to provide a means to allow mining agents to opportunistically reserve a portion of the set of files to be mined so that agents can collaborate on the mining of multiple years worth of data, with each agent being able to reserve the next available unmined year.

Acknowledgement

We would like to acknowlege the debt owed to individuals at the University of Alabama in Hunstville for the development of ADaM. While one of the authors of this paper, Hinke, designed an original data mining system that was a precursor to the ADaM data mining system, Dr. John Rushing was the principle architect and implementor of the ADaM data mining system. Other significant contributors to the NASA grant that produced ADaM, were Dr. Sara Graves and Dr. Heggere Ranganath.

References

1. Johnston, B., D. Gannon, and B. Nitzberg. *Grids as Production Computing Environments*. in *8th IEEE Symposium on High Performance Distributed Computing*. 1999.

2. *NASA Workshop on the Issues in the Application of Data Mining to Scientific Data*. www.cs.uah.edu/NASA_Mining. 1999.

3. Hinke, T., J. Rushing, S. Kansal, S. Graves, and H. Ranganath. *For Scientific Data Discovery: Why Can't the Archive be More Like the Web*. in *Proceedings Ninth International Conference on Scientific Database Management*. 1997.

4. Hinke, T., J. Rushing, H. Ranganath and S. Graves. *Techniques and Experience in Mining Remotely Sensed Satellite Data.* Artificial Intelligence Review: Issues on the application of data mining, 2000. In press.

5. Hinke, T., J. Rushing, H. Ranganath and S. Graves.. *Target-Independent Mining for Scientific Data*. in *Proceedings: The Third International Conference of Knowledge Discovery & Data Mining*. 1997.

6. *Special Sensor Microwave/Imager data*. Captured by a Defense Meteorological Program satellite and obtained from the Global Hydrology Resource Center, Huntsville, AL.